

**Amendments to the Claims:**

This listing of the claims replaces the listings of the claims in the present patent application:

**Listing of Claims:**

1. **(Currently Amended)** A method for displaying an animation, comprising:
  - receiving an instruction to display an animation, the animation comprising a plurality of images ordered for sequential display;
  - retrieving an animation file responsive to the instruction, the animation file providing an ordering of the images;
  - determining a maximum size, the maximum size, related to a maximum amount of memory usable to load images;
  - determining a first set of images, which aggregate to a size up to the maximum size, the first set of images having final image;
  - determining a second set of the images, which aggregate to a size up to the maximum size, the images in the second set being in sequence behind the final image;
  - generating a first segment file indicative of the first set of images, ~~the first set of images not linked to each other;~~
  - generating a second segment file indicative of the second set of images, ~~the second set of images not linked to each other;~~
  - associating a callback identifier with the second segment file;
  - providing the callback identifier along with the first segment file;
  - loading the first set of images into a memory readable by an animation engine according to the first segment file;
  - displaying sequentially, using the image order in the animation file, each image in the first set as a first animation segment;
  - retrieving the callback identifier from the first segment file;
  - using the callback identifier to load the second set of images into the memory according to the second segment file; and

displaying sequentially, using the image order in the animation file, each image in the second set as a second animation segment.

2. **(Original)** The method for displaying an animation according to claim 1, wherein each of the images are stored as individual graphics files.

3. **(Original)** The method for displaying an animation according to claim 1, wherein the animation file further comprises information indicative of the size of individual ones of the images, and the size information is used in determining the first set of images.

4. **(Previously Presented)** The method for displaying an animation according to claim 1, wherein the maximum size is set to further correspond to a number of images.

5. **(Previously Presented)** The method for displaying an animation according to claim 1, wherein the maximum size is set at a predetermined memory size for an embedded system.

6. **(Original)** The method for displaying an animation according to claim 1, wherein the maximum size is generated responsive to an inquiry regarding available memory.

7. **(Original)** The method for displaying an animation according to claim 1, wherein the first segment file provides a file identifier for each of the images in the first set.

8. **(Original)** The method for displaying an animation according to claim 1 wherein the associating step includes using the callback identifier as a name for the second segment file.

9. **(Original)** The method for displaying an animation according to claim 1 wherein the associating step includes placing the callback identifier as data in the second segment file.

10. **(Original)** The method for displaying an animation according to claim 1 wherein providing the callback identifier includes using the callback identifier as part of a name for the first segment file.

11. **(Original)** The method for displaying an animation according to claim 1 wherein providing the callback identifier includes placing the callback identifier as data in the first segment file.

12. **(Original)** The method for displaying an animation according to claim 1 wherein the second set of images are being loaded into the animation processor while the images in the first set are being displayed.

13. **(Currently Amended)** A method of generating animation segment files, comprising:

- receiving an animation file that identifies and orders a set of images;
- dividing a maximum size, the maximum size related to a maximum amount of memory usable to load images;
- dividing the set of images into sequential subsets of images, each subset having a size up to a maximum size, ~~where individual images have no link to another image,~~ being indicative of an animation segment;
- associating a subset identifier with each respective subset;
- associating an action instruction with each respective segment;
- wherein the action instruction associated with one subset identifies another one of the subsets; and
- displaying sequentially each image in the animation segment.

14. **(Previously Presented)** The method of generating animation segment files according to claim 13, wherein the maximum size further corresponds to a number of images of known size.

15. **(Previously Presented)** The method of generating animation segment files according to claim 13, wherein the maximum size is set to a memory size derived from an amount currently usable for loading images.

16. **(Original)** The method of generating animation segment files according to claim 13, wherein an action instruction is used to identify the last subset.

17. **(Currently Amended)** A method of displaying an animation, comprising:

- receiving an instruction to display the animation, the animation comprising a set of sequential images;

- retrieving a first segment file, the first segment file identifying a first subset of the images, ~~the first subset of images not linked to each other;~~

- loading the first subset of images and sequentially displaying the images in the first subset of images;

- extracting a callback instruction using the first segment file, the callback instruction identifying a second segment file;

- releasing memory holding at least one of the images in the first subset of images;

- retrieving the second segment file, the second segment file identifying a second subset of the images, ~~the second set of images not linked to each other;~~ and

- loading the second subset of images and sequentially displaying the images in the second subset of images.

18. **(Original)** The method of displaying an animation according to claim 17, wherein loading the second subset is initiated before all the images in the first subset have been displayed.

19. **(Cancelled)**

20-23. **(Cancelled)**

24. **(Currently Amended)** A method for sequencing a plurality of media objects into a presentation, comprising:

- receiving a media file providing a first one of the media objects and a second one of the media objects, the second media object being an animation file;

- associating a callback identifier with the second media object;

- providing the callback identifier along with the first media object;

- loading the first media object into a memory usable for presenting the first media object;

- using the callback identifier to load the second media object into the memory that is usable for presenting the second media object where the animation file has at least a first and second segment associate with it, each segment comprising at least one image, each image being a displayable image, ~~the first segment identifying a first set of images not linked to each other if there are a plurality of images, the second segment identifying a second subset of images not linked to each other if there are a plurality of images,~~ where each segment has a size up to a maximum size, the maximum size being related to a maximum amount of memory usable for loading images.

25. **(Previously Presented)** The method of claim 24, wherein the first media object is a sound file.

26 - 27. **(Cancelled)**

28. **(Previously Presented)** The method of claim 24, further including a third one of the media objects, the third media object having an action instruction indicative of a duration to present the third media object.

9. **(Previously Presented)** The method of claim 24, wherein the first media object has an action instruction for loading a third one of the media objects, the third media object enabled for presentation concurrently with a first media object.

30. **(Previously Presented)** The method of claim 29, wherein the third media object is enabled for presentation after the second media object.